

# WP-CLI Cheatsheet

Quick `wp` reference for real operations: content, SEO, maintenance, users, and automation. For creators and operators who work with WordPress every day.

## Content

List, create, update and delete posts without touching the admin.

```
# List posts by status
wp post list --post_status=publish --fields=ID,post_title,post_date

# Create a post from CLI
wp post create --post_title="My article" \
  --post_status=draft \
  --post_content="Content..." \
  --post_author=2

# Update multiple posts
wp post update 123 456 789 --post_status=publish

# Delete a post (to trash, recoverable)
wp post delete 123
# Delete permanently
wp post delete 123 --force

# Create a page
wp post create --post_type=page --post_title="About" \
  --post_status=publish --post_content="..."

# Duplicate a post (clone content + title)
wp post create --post_title="Copy of X" --post_status=draft \
  --post_content="$(wp post get 123 --field=post_content)"

# List posts modified in the last 7 days
wp post list --post_modified="$(date -d '7 days ago' '+%Y-%m-%d')" --fields=ID,post_title

# Replace a string across multiple posts
wp search-replace 'old-text' 'new-text' wp_posts --dry-run
wp search-replace 'old-text' 'new-text' wp_posts --all-tables

# Change the author of a post
wp post update 123 --post_author=5

# Empty the trash
wp post delete $(wp post list --post_status=trash --format=ids)
```

## SEO and metadata

Work with Yoast/RankMath and custom metadata.

```
# View SEO title of a post (Yoast)
wp post meta get 123 _yoast_wpseo_title

# Update meta description
```

```

wp post meta update 123 _yoast_wpseo_metadesc "Description < 155 chars"

# List all posts without meta description
wp post list --meta_key=_yoast_wpseo_metadesc --meta_compare=NOT EXISTS \
  --post_type=post --post_status=publish --format=ids

# Update focus keyword in bulk
for id in $(wp post list --post_type=post --post_status=publish --format=ids); do
  wp post meta update $id _yoast_wpseo_focuskw "keyword-{$id}"
done

# Export all titles + slugs to CSV
wp post list --post_type=post --post_status=publish \
  --fields=ID,post_title,post_name --format=csv > posts.csv

# Regenerate slugs from titles
wp eval 'foreach
(get_posts(["post_type"=>"post","post_status"=>"publish","posts_per_page"=>-1]) as $p)
{ wp_update_post(["ID"=>$p->ID, "post_name"=>sanitize_title($p->post_title)]); }'

# Sitemap
wp rewrite list | grep sitemap
wp sitemap generate --help

```

## Maintenance

*Updates, plugins, themes, database.*

```

# WP version
wp core version
wp core check-update

# Update core
wp core update
wp core update-db

# List plugins with status and version
wp plugin list

# Update all plugins
wp plugin update --all

# Deactivate + uninstall a plugin
wp plugin deactivate akismet && wp plugin uninstall akismet

# List themes
wp theme list

# Clean revisions
wp post delete $(wp post list --post_type=revision --format=ids) --force

# Cap future revisions to 5
wp config set WP_POST_REVISIONS 5 --type=constant

# Optimize the database
wp db optimize
wp db repair

# DB size

```

```
wp db size --tables
wp db query "SELECT table_name, ROUND(data_length/1024/1024, 2) MB FROM
information_schema.tables WHERE table_schema=DATABASE() ORDER BY data_length DESC;"

# Cron
wp cron event list
wp cron event run wp_scheduled_delete
wp cron test
```

## Users

*User, role and capability management.*

```
# List users
wp user list --fields=ID,user_login,user_email,roles

# Create user
wp user create jane jane@example.com --role=editor --user_pass="StrongPass123!"

# Change role
wp user set-role 5 author

# Reset password (prompts interactively)
wp user update jane --user_pass="NewPass123!"

# Generate and show a password
wp user update jane --user_pass=$(openssl rand -base64 16)

# Delete user and reassign content
wp user delete jane --reassign=2

# List capabilities of a role
wp cap list editor

# Add a custom capability to a role
wp cap add editor edit_theme_options

# Active sessions
wp user session list jane
wp user session destroy jane all
```

## Config (wp-config.php)

*Constants, salts, debug.*

```
# Read a constant value
wp config get WP_DEBUG
wp config get WP_HOME

# Set (type=constant, type=variable, type=site-option)
wp config set WP_DEBUG true --raw --type=constant
wp config set WP_DEBUG_LOG true --raw --type=constant
wp config set WP_DEBUG_DISPLAY false --raw --type=constant

# Enable debug temporarily
wp config set WP_DEBUG true --raw --type=constant
wp config set WP_DEBUG_LOG true --raw --type=constant
```

```

wp config set WP_DEBUG_DISPLAY false --raw --type=constant

# Disable
wp config delete WP_DEBUG
wp config delete WP_DEBUG_LOG
wp config delete WP_DEBUG_DISPLAY

# Regenerate salts
wp config shuffle-salts

# Move domains
wp search-replace 'https://old.com' 'https://new.com' --dry-run
wp search-replace 'https://old.com' 'https://new.com' --all-tables

# Force HTTPS in HOME and SITEURL
wp option update home 'https://d0a1.es'
wp option update siteurl 'https://d0a1.es'

```

## Automation with cron

*Schedule recurring tasks inside and outside WordPress.*

```

# List WP cron jobs
wp cron event list
wp cron schedule list

# Schedule a custom hook
wp cron event schedule my_custom_hook now hourly \
  --args='["param1","param2"]'

# Run a hook now
wp cron event run my_custom_hook

# System crontab (crontab -e)
# Daily DB backup at 3am
0 3 * * * cd /var/www/html && wp db export /backups/db-$(date +%F).sql
# Regenerate thumbnails weekly
0 4 * * 0 cd /var/www/html && wp media regenerate --yes
# Clean spam every hour
0 * * * * cd /var/www/html && wp comment delete $(wp comment list --status=spam --
format=ids) --force

# Remote WP-CLI via SSH alias
wp @production post list --ssh=wp@d0a1.es:/var/www/html
wp @production db export backup.sql

```

## WP-CLI with d0a1

*From inside the Docker container:*

```

# Enter the wordpress container
docker compose exec wordpress bash

# One-liner
docker compose exec wordpress wp post list --post_status=publish

# Full backup (DB + uploads)

```

```
docker compose exec wordpress wp db export - > backup-$(date +%F).sql
docker compose exec wordpress tar czf - /var/www/html/wp-content/uploads > uploads-$(date +%F).tar.gz
```

**More at** [d0a1.es/en/wp-cli-cheatsheet-en/](https://d0a1.es/en/wp-cli-cheatsheet-en/) · [Full docs](#) · [hello@d0a1.es](mailto:hello@d0a1.es)